

Semantic classifier approach to document classification

Piotr Borkowski, Krzysztof Ciesielski, and Mieczysław A. Kłopotek

Institute of Computer Science, Polish Academy of Sciences,
ul. Jana Kazimierza 5, 01-238 Warszawa, Poland
Tel.: (+48) 22 380-05-00
Fax: (+48) 22 380-05-10
piotrb, kciesiel, klopotek@ipipan.waw.pl

Abstract. In this paper we propose a new document classification method, bridging discrepancies (so-called semantic gap) between the training set and the application sets of textual data. We demonstrate its superiority over classical text classification approaches, including traditional classifier ensembles. The method consists in combining a document categorization technique with a single classifier or a classifier ensemble (SemCom algorithm - Committee with Semantic Categorizer).

1 Introduction

The text document classification methods are well-established in the area of text mining. Predominantly they have been derived from corresponding data mining techniques that were designed to handle long input data records. Let us mention here for example Naive Bayes, Balanced Winnow and LLDA (to be described later). While these methods are quite successful in data mining and were appreciated within text mining community, one important drawback occurs related to the specific area of text mining. While in data mining the meaning and the value range of individual attributes of an object are relatively well defined, in text mining it is not the case any more. Same content may be expressed in different ways, using different words (via synonyms, list of hyponyms) while the same word can express different things in different contexts. This would not be a big obstacle if not the fact that traditional techniques would require significantly larger bodies of training data, which makes an unbalanced sample much more likely. Not only because of the size of the data sample but also the heterogeneity of the data sources that need to be combined. It is even worse when the trained classifiers need to be applied to unseen data which stems from a dataset that from the human point of view touches the same topic but from the computer point of view is written in a completely different style. This gives rise to so-called *semantic gap*, that is though the training and application data sets are semantically similar, their syntactical and bag-of-words view differ. In such a case understanding the semantics of documents would be needed, which is unavailable for traditional data mining techniques.

In this paper we propose two new document classification methods, SemCla (Semantic Classifier) and SemCom (Committee with Semantic Categorizer), bridging the semantic gap between the training set and the application sets of textual data. The methods consist in combining an unsupervised *document categorization technique* with a

single classifier or a classifier ensemble. Via this component the traditional notion of document similarity (based on angles between vectors in term space) is amended to include the concept of *semantic similarity*. The notion of semantic similarity, as used in this paper, was described in [1]. Both methods introduced in the paper are based on our SemCat (Semantic Categorizer) algorithm, that has also been introduced in [1].

In Section 2 we define the problem of document categorization and semantic classification and recall the work done on the subject by other researchers. In Section 3 we describe our categorization methodology, SemCat. Subsequently we show in Section 4, how our categorization method can be used in various ways in the classical task of classification.

In Section 5 we explain the setup of experiments we performed to show the usefulness of SemCla algorithm in classification tasks. In subsequent Section 6 showing the results of these experiments, we demonstrate superiority of the semantic classification methods (SemCom and SemCla) over classical text classification approaches, including traditional classifier ensembles for text classification tasks (Section 6.1) as well as in cases when the so-called semantic gap occurs (Section 6.2).

Section 7 summarizes achieved results and outlines future research directions.

1.1 Our contribution

Our contribution in this paper is:

- constructing new supervised classifier based on unsupervised semantic document categorizer,
- demonstrating feasibility of the new classifier for bridging semantic gap between test and training set of data,
- designing a heterogeneous committee that combines classical classifiers and the semantic classifier.

2 Previous work

The task of *categorization* is to assign one or more labels (categories) to a document, or a group of documents (cluster labeling). It finds multiple practical applications, especially for assisting in text retrieval task: in web page classification, e-mail and memo organization, expanding queries with new terms, expanding / improving ontologies, and many other.

The categorization task can be viewed formally as a special case of classification [2,3], but with a couple of differences. First of all, the number of categories significantly exceeds the number of classes in typical classification task. Categories may be flat and disjoint, but they may form a tree or even a hierarchy (acyclic graph). And more than one category may be assigned to a single document. Therefore typical classification methods do not fit well to the task of categorization. Diverse other methods have been proposed to attack the problem of categorization. Some of them are based on clustering. The most popular representatives of this brand of approaches are Nonnegative Matrix Factorization (NMF), Latent Semantic Analysis (LSA), Probabilistic LSA (PLSA),

and Finite Mixture of Multidimensional Bernoulli Distributions, described in [4]. Other researchers map the document contents to some semantic resources, in particular to Wikipedia (\mathfrak{W}). This approach was exploited in *WikipediaMiner Project*¹, developed at the University of Waikato in Hamilton, New Zealand [5,6,7]. It uses \mathfrak{W} topics as categories. Basic idea was key phrase indexing. For terms from \mathfrak{W} their “keyphraseness” [8] that is share of occurrences in \mathfrak{W} links is computed. Then these terms are searched in a document to be categorized. Terms with multiple meanings are disambiguated (via some trained classifier) by choosing the meaning most close to the document topic. For training purposes documents annotated with such keyphrases have to be assigned categories. Then a classifier is trained.

In this paper we exploit our new unsupervised categorization method, SemCat, introduced in [1]. Contrary to WikipediaMiner, no classifiers are used, hence no training corpora need to be prepared. Also it is not based on \mathfrak{W} links. Instead the category graph of \mathfrak{W} is exploited. A novelty here is also the usage of more challenging Polish language [9]. Furthermore, we develop a classification method SemCla suitable to apply for data with semantic gap.

The problem of “semantic gap” is understood in literature in many ways. We focus on the aspect encountered in text retrieval where data come for different domains. The next paragraphs give a brief overview of the approaches that have been proposed.

The article [10] shows a review of *cross-domain text categorization* problem. Unlike the classical case, the training and the test data originates from different distributions or domains. This is very common in practical tasks because (especially for Polish language) we often do not have a suitable data set of labeled documents. Often what we have is a corpus which is topically related, but presents the same (or semantically similar) information in a different way, e.g. using different vocabulary. Many algorithms have been developed or adapted for cross-domain text classification, there are conventional algorithms: Rocchio’s Algorithm, Decision Trees like: CART, ID3, C4.5; Naive Bayes classifier, KNN, Support Vector Machines; and some novel cross-domain classification algorithms: Expectation-Maximization Algorithm, Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation(LDA), CFC Algorithm, Co-cluster based Classification Algorithm [11].

Paper [12] gives a general overview of the problem of semantic gap in information retrieval. Authors focus on two separate task: text and multimedia mining/image retrieval. Semantic gap in text retrieval is defined as a usage of different words (synonyms, hypernyms, hyponyms) to describe the same object. In the part about text retrieval authors concentrate on reorganizing search results by using post-retrieval clustering system. They work on search results (“snippets”) and enhance them by adding so called *topics*. Topic is a set of words (they have similar meaning) that was as outcome of Probabilistic-Latent Semantic analysis or Latent Dirichlet Allocation on some external data collection. After adding a topic to the snippet they carry out clustering or labeling.

In the paper [13] authors propose a way to improve categorization by adding semantic knowledge from Wikitology (knowledge repository based on Wikipedia). They used various text representation and text enrichment techniques and used Support Vector Machine-SVM to learn a model of classification.

¹ <http://wikipedia-miner.sourceforge.net/>

3 Our taxonomy-based semantic categorization method

Our taxonomy-based categorization method SemCat was described in detail in [1], so below we present only its brief description.

3.1 Outline of the algorithm

Suppose we have a taxonomy of categories (a directed acyclic graph with one root category) like Wikipedia (\mathfrak{W}) category graph or *Medical Subject Headings* (MeSH) ontology². We assume there is a set of concepts connected with the taxonomy, in the following way: every concept is linked to one or more categories. Every category and concept is tagged with a string label. Strings connected with categories are used as an outcome presented to a user. And those attached to concepts are used for mapping a text of document into the set of concepts.

For the experimental design we used \mathfrak{W} category graph with the concept set of \mathfrak{W} pages. Tags for \mathfrak{W} categories were their original string names. Set of string tags connected with a single \mathfrak{W} page consists of: lemmatized page name and all names of disambiguation pages that link to that page.

In the process of document categorization we remove stop words and very rare / frequent words, lemmatize, find phrases and calculate normalized *tfidf* weights for terms and phrases. Calculation of a standard *term frequency inverse document frequency* is based on word frequencies from the collection of all \mathfrak{W} pages.

Then we map document's terms and phrases into a set of concepts. In the case of homonyms, we disambiguate the concept assignment: we select the concept that is the nearest by similarity measure defined by Equations (1) and (2) (see Section 3.2) to the set of concepts that was mapped in an unambiguous way. We investigated other methods of disambiguating e.g taking all meanings of ambiguous terms and weigh them accordingly. The results for various disambiguation methods are described in Section 5.4.

When every term in the document is assigned to a proper concept (\mathfrak{W} page), then all concepts are mapped to \mathfrak{W} categories. In this way usually one term maps to more than one category, so we transfer the weight associated to that term proportionally to all its categories. Sum of weights assigned to the categories equals to sum of *tfidf* for terms. The outcome of that procedure is a ranked list of categories with weights. In the last step we can transform the weighted ranking and / or choose top-*N* categories out of it.

3.2 Similarity measures

We use semantic measures for matching concepts (\mathfrak{W} pages) and objects of the taxonomy (\mathfrak{W} categories). We were inspired by the paper [14]. The semantic measures are based on: the unary function *IC* (*Information Content*) and binary function *MSCA* (*Most Specific Common Abstraction*). Their inputs are categories from a taxonomy.

² <https://www.nlm.nih.gov/mesh/>

Though superficially similar, our IC definition differs essentially from that proposed for WordNet. WordNet computes the IC for concepts based on the number of subordinated concepts. We compute the IC for categories, based on the count of concepts that belong to subordinated categories. So the IC of a category is weighted by the frequency of its usage in the language rather than by its definitional complexity.

For a given category k we define $IC(k) = 1 - \log(1 + s_k) / \log(1 + N)$, where s_k is the number of taxonomy concepts in the category k and all its subcategories, and N is the total number of taxonomy concepts. The main category has the lowest value of $IC = 0$.

For two given categories k_1 and k_2 we define $MSCA(k_1, k_2)$ as the category $k^* \in CA(k_1, k_2)$ (the set of super-categories for both categories k_1 and k_2) that maximize a value of the function $\{IC(k) : k \in CA(k_1, k_2)\}$. The properties of $IC(\cdot)$ measure ensure that the category chosen is most specific amongst the common super-categories.

In the literature dealing with Wordnet many measures based on IC and $MSCA$ have been proposed [14], including LIN and PIRRO-SECO similarity:

$$sim_{Lin}(k_1, k_2) = \frac{2 \cdot MSCA(k_1, k_2)}{IC(k_1) + IC(k_2)} \quad (1)$$

$$sim_{PirroSeco}(k_1, k_2) = \frac{1}{3} \left(3 \cdot MSCA(k_1, k_2) - IC(k_1) - IC(k_2) + 2 \right) \quad (2)$$

Though analogous measures were defined for WordNet, our category similarity measures differ from those for WordNet because we defined IC and MSCA differently than in Wordnet. Our definition is based on Wikipedia structure, hence we do not need to refer to WordNet.

We used the above measure for categories to define a similarity measures for concepts (\mathcal{W} pages). Similarity between pages p_i and p_j is computed by aggregation of the similarity between each pair of categories (k_i, k_j) such that p_i belongs to the category k_i and p_j to k_j :

$$sim_{PAGE}(p_i, p_j) = \max\{sim_{CAT}(k_i, k_j) : p_i \in k_i \wedge p_j \in k_j\} \quad (3)$$

4 Application to classification task

In order to demonstrate the value of semantic categorization, we exploited it as an ingredient (to a classifier ensemble) in the classical classification algorithms and their committees, SemCom, as well as an stand-alone classifier SemCla.

In this section we recall commonly known classification algorithms we used in our experiments. These were Naive Bayes, Balanced Winnow, Labeled LDA, as well as the committees of classifiers (bagging type ensembles) built upon Naive Bayes classifier and Balanced Winnow. We describe also our own semantic categorization based classifier SemCla and our heterogeneous committee SemCom (containing both proprietary SemCat method and above-listed supervised classification methods).

4.1 Naive Bayes

Naive Bayes classification method (cf. [15]) on the basis of knowledge derived from training data set, creates a probabilistic model assigning one of the predefined classes (i.e. labels) to a new observation (i.e document). In this approach, each document is treated as a bag of words, which does not take into account the order (syntax). Additionally, a simplifying assumption is made, that the individual words in the document are independent. The probability of a given class c being assigned to a document d is calculated as follows:

$$P(c|d) = \frac{P(c) \prod_{w \in d} P(w|c)^{n_{wd}}}{P(d)},$$

where n_{wd} is the total number of occurrences of word w in the document, a $P(w|c)$ is the probability of occurrence of a word w in the class c . $P(c)$ is the probability of the class c which is estimated based on the fraction of documents that belongs to this class. The value of $P(d)$ does not depend on the class, thus it is ignored for the purpose of document classification. Finally, $P(w|c) = \frac{1 + \sum_{d \in D_c} n_{wd}}{k + \sum_{w'} \sum_{d \in D_c} n_{w'd}}$, where D_c is the set of all documents in the class c , and the number of k is the size of the dictionary (i.e. the number of distinct words).

4.2 Balanced Winnow

Balanced Winnow algorithm details can be found in [16] and [17]. Several versions of this classifier can be found in the literature. Main concept is based on the Perceptron algorithm (cf. [18]). For our purpose Balanced Winnow version of the algorithm was selected because of its high observed efficacy. For each word, algorithm stores two weights: w^+ and w^- , on the basis of which algorithm calculates document membership to each class (binary classification). Positive weights are in favor of a given class, negative weights against it. The difference between the weights ($w^+ - w^-$) is the overall weight associated with a given word. Assume that the classified document is a vector of words with the weights $x = (x_1, \dots, x_n)$. Then the classification rule is based on the inequality $\sum_{i=1}^n (w^+ - w^-) x_i > \theta$, for a fixed value of the parameter θ . Training of the classifier is based on weights modification, but only if the training document has been misclassified. Two parameters are introduced: promotion level $\alpha > 1$ and degradation level $0 < \beta < 1$. If the error is to classify the document to the class to which it does not belong (negative document), then the weights of the words are modified as follows: $w^+ := \beta w^+$, $w^- := \alpha w^-$. If an error is made on a positive document (by not classifying it to the positive class), weight modification is as follows: $w^+ := \alpha w^+$, $w^- := \beta w^-$.

4.3 Labeled LDA

Labeled Latent Dirichlet Allocation (LLDA) is an extension of the popular – among practitioners and theorists – Latent Dirichlet Allocation model described in [19]. It is one of many probabilistic topic models useful in analyzing text documents. In particular the review of this subject can be found in [20].

LDA is an unsupervised method, where any document is treated as a probabilistic mixture of various topics. Resulting generative model is characterized by the discrete probability distribution of words within a given topic. The model assumes the following way to generate each document. The length N of the document is selected (the Poisson distribution is used). Then the proportion of subjects making up the document is fixed (Dirichlet distribution randomizing the set of K topics). Subsequent words in the document are generated by the random selection of the topic (with a multinomial distribution generated above), and then within this topic (determining the distribution of words), a particular word is generated. Assuming such a method of generating each document in a given collection, LDA is trying to recreate a set of topics that are generating the observed collection. Labeled LDA method is a supervised variant, which relates every document label to a fixed subset of topics. LLDA algorithm is very similar to its unsupervised prototype, with the exception that the document topics are selected only from among those that correspond to the observed document labels – details can be found in [21]. There are other supervised variants of the LDA algorithm, such as Supervised LDA ([22]). Authors selected LLDA in favor of Supervised LDA since in our experimental settings LLDA gave significantly better results. As part of future work it is planned to use also semi-supervised methods such as Partially Labeled Dirichlet Allocation (cf. [23]).

4.4 Semantic classification

Below we present a description of a new *semantic classifier* which we call SemCla. It is based on a category representation of a document produced by SemCat (see Section 3.1), which is used in combination with semantic measures (see Section 3.2).

Outline of the algorithm Recall that SemCat uses words and phrases from the document to produce a list of categories with weights. This representation of a document can be considered as a vector of weights for all category from \mathfrak{M} category structure. Therefore we call it *vector of categories*. We use it to calculate cosine product. We found out that the algorithm performs better when for each category from the vector of categories we add a super category of it (according to \mathfrak{M} hierarchy) with weight equal the initial weight multiplied by a constant α (we used the value $\alpha = 0.33$, we explain below how we calibrated this parameter). Thus we obtain the *extended category vector*. This process is visualized in Figure 1.

The semantic classification is made in the way described below and illustrated in Figure 2.

1. documents from training and test sets are categorized to obtain category vectors that represent their content,
2. category vectors for all documents are changed into extended category vectors (for constant α),
3. we classify a new document (represented by its extended category vector) by finding the nearest group (in the sense of the cosine product) in the training set.

In the literature, a group to be compared with is represented by its centroid. Although the method with centroids works faster, it gives poorer results. Therefore results presented in Tables 1 – 4 are for SemCla algorithms that find the nearest group using all documents from the group and taking average similarity.

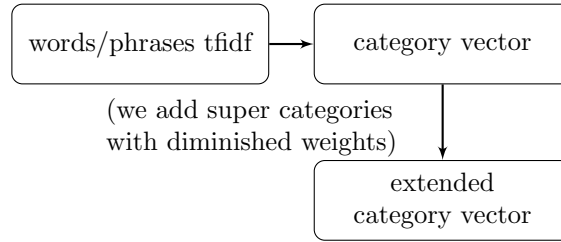


Fig. 1. Single document category representation

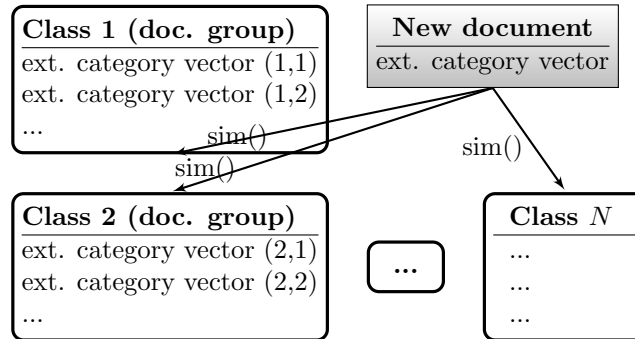


Fig. 2. Categorization as a classification (SemCla algorithm)

Finding optimal α parameter Value of optimal α was found in a separate experiment before the experiment discussed in the paper. It was conducted for SemCat algorithm. We took 4 groups of documents from *kopalniawiedzy.pl*: *astronomy-physics*, *psychology*, *medicine*, *technology* and drew at random $N = 100$ documents from each of it. We did not use all document groups from this corpus, we chose 4 groups that were most different from each other. All documents were categorized with various values of α ranging from 0.0 to 0.5 (bigger α resulted in a significant deterioration of the outcomes). Then we calculated semantic similarity between categorized document (with different α), sorted them and ranked. We chose the value of parameter α that maximizes difference between means of rank of documents from the same groups and those

belonging to different groups. In other words, we found the value that separates best these groups of documents.

4.5 Ensemble of classifiers

The experimental setting was also based on the ensemble of classifiers. For each document the classification process is carried out by every classifier in the ensemble (it may also be a classifier of the same type, but trained on a different learning sample). Then the results of all classifiers are aggregated as the final ensemble classifier. In the existing implementation this can be done in three ways: (a) each classifier has one vote – category with the highest number of votes is selected; (b) votes counting additionally takes into account the weights of classification results (this option requires that all classifiers are of the same type); (c) ranks of the elements returned by the classifier are aggregated instead of raw votes or weights. In the case when two (or more) categories received exactly the same number of votes, the result is selected at random from among the winning categories.

4.6 Heterogeneous committee of classifier with categorization method

In our new approach, we developed heterogeneous committee of classifiers SemCom that contains the supervised methods of Naive Bayes, Balanced Winnow, LLDA and our proprietary unsupervised categorization method SemCat utilizing taxonomy of \mathcal{W} categories.

Categorization method is unsupervised, and thus it cannot be trained on different samples in a similar manner to supervised classifiers (categorization method utilizes data from the complete \mathcal{W} taxonomy). For this reason the committee contained only one instance of the categorization algorithm. In order to increase the impact of SemCat on the final results of the committee as a whole, categorization votes were counted with the higher weight. In addition, one should take into account that the categorization algorithm returns a ranking of categories (not only a single category). Thus, in the experimental settings we included a variant of the committee in which categorization method add more than one category with the highest rank in the list (and the correspondingly decreasing weights).

4.7 Remarks on denotation of classifier and ensemble parameters and composition

Experimental setting exploited several variants of the ensembles, trained on a different subsets from the training set (\mathcal{W} pages for Table 1, 2 and groups of news for Tables 3, 4).

For the *classical classification task* (Table 1 and 2) from all \mathcal{W} pages belonging to a single category [$S = 50, 100, 200$] pages were drawn at random and on the basis of such a sample, a single classifier was trained. For a given set of classes, into which documents are to be classified, we choose \mathcal{W} categories that represent these classes. We

will call them \mathfrak{W} class categories. When we choose \mathfrak{W} documents for training, we can choose either documents that have \mathfrak{W} categories identical with the \mathfrak{W} class categories or their sub-categories. We say that we choose level 1 ($L = 1$) documents, if for each document at least one of its categories is identical with class category. If we choose $L = 2$ documents, then we choose additionally also documents that have categories being direct subcategories of the class category. Vector of numbers following the SemCat represents weights attached to top-3 categories inserted into committee. We chose them among all categories produced by SemCat algorithm (e.g SemCat:(7,5,3) means that we put top three categories from semantic categorizer with weights 7, 5 and 3). For the *semantic gap task* we used $S = 50$ for Table 3 and $S = 200$ for Table 4. Experimental committees consisted of 25 classifiers based on Naive Bayes and Balanced Winnow methods. Aggregation variant was the one in which each classifier is voting on one category only. More information on ensemble methods can be found in [24].

5 Experimental setup

5.1 Performed experiments

We performed two types of experiments, their results are reported in Tables 1 – 4. The first experiment aimed at demonstrating that adding a semantic categorizer to a committee of traditional classifiers improves the classification correctness in *classic classification task* (Table 1, 2). The second experiment was designed to show that a semantic categorizer is capable of *bridging semantic gap* between the training data and the test data (Table 3, 4).

5.2 Benchmark data sets

For experimental purpose we used two different benchmark data sets. We needed different datasets because of various nature of the investigated problems.

Benchmark used for classification comparison.

The benchmark data set was based on Polish subdirectory of DMOZ taxonomy / Open Directory Project <http://www.dmoz.org>. It contains 1063 text files of Polish web pages just with html tags removed. Selected documents belong to 15 directories that map into \mathfrak{W} categories. They are: astronomy, biology, economics, philosophy, physics, graphics, history, linguistics, mathematics, education, politics, law, religious studies, sociology, technology. None of these categories is a subcategory of another one in the \mathfrak{W} taxonomy. We omitted a few cases of multi-labeled documents. For the benchmark documents the reader is referred to the benchmark web page³. The various options of categorization setting cause the number of categorized document differs. For calculating the results we choose a set of documents that was categorized by every algorithm.

Benchmark containing data with semantic gap.

³ http://www.ipipan.waw.pl/~kciesiel/iis/DMOZ_PL_taxonomy.zip

The second benchmark was made of documents downloaded from various news page. It consists of training and evaluation part, they come from various domains. We used separate collections to achieve different wordings in each of them. The training set consists of news from the popular science portal `kopalniawiedzy.pl` merged with documents from one directory from `forsal.pl` – the domain about finance and economy. Below we show more detailed description of the training set.

- documents from `kopalniawiedzy.pl`: **astronomy-physics** N=283; **medicine** N=2979; **life science** N=3122; **technology** N=4861; **psychology** N=1733; **humanities** N=244,
- documents from `forsal.pl` from the directory **Gielda (Stock exchange)** N=1987.

For evaluation we downloaded directories from `www.rynekzdrowia.pl` (containing medical news) and merged it with economical documents from `www.forsal.pl` and `www.bankier.pl` (market, finances, business). Datasets used for evaluation:

- directories from `www.rynekzdrowia.pl`: **Ginekologia (Gynecology)** N=1034; **Kardiologia (Cardiology)** N=239; **Onkologia (Oncology)** N=1195,
- directories from `www.forsal.pl`: **Waluty (Currencies)** N=2161; **Finanse (Finances)** N=1991,
- documents from `www.bankier.pl` N=978.

5.3 Efficiency measures

To assess the efficiency of the studied algorithms we use two different measures. The first one is commonly used standard *precision* measure, the second one is modified precision based on similarity measure *Lin* (Equation (1) in Section 3.2). The difference is in using *Lin* measure instead of indicator function. For documents d_1, \dots, d_n with real categories $categ(d_i)$ and its prediction $pred(d_i)$ the *Lin precision* is defined as: $\frac{1}{n} \sum_{i=1}^n Lin(categ(d_i), pred(d_i))$. The motivation for using the latter measure is that standard precision does not take into account the dependence between categories. In case when we make a wrong prediction we would like to know how much predicted category is different from the real one.

5.4 Classical classification task

The first part of experimental work concerned comparison of various methods of text classification. We proceeded on documents from DMOZ corpus with fixed set of labels described in Section 5.2. Documents were divided into separate groups based on their text length measured by the number of characters (C): *short* ($1000 \leq C < 2000$), *medium* ($2000 \leq C < 10,000$), *long* ($10,000 \leq C$). Files shorter than 1000 characters were not processed. Results for various classification methods are presented in Tables 1, 2. They were divided by a file size and efficiency measure. Methods based on categorization algorithm return a list of weighted \mathfrak{M} categories. Therefore we transformed the outcome categories into the target set of 15 categories and took only one category with the highest weight. Categorization was based on a selection of 10 words (only nouns) / phrases with highest *tfidf* from the document. The experiments were performed for different values of parameters, but other settings gave worse results.

In Table 1 first four rows present various modifications of categorization method. The difference between them is in the method of disambiguation of ambiguous page. The first row presents standard disambiguation method (see Section 3). The next two methods find a set of pages that map unambiguously. Then for every ambiguous page we find all of its mappings to potential meanings. Then we figure out their distances to the set and sort them into descending order. Subsequent possible meanings are given various weights depending on their rank i : ($1/2^i$, $1/i$ or uniformly). All of these options gave similar means, so we used paired t-test to compare them. As a reference we used basic disambiguation method. Methods with weighting $1/2^i$ and $1/i$ do not differ significantly. Method with uniform weights differs.

All of these methods took only nouns from the document. We developed two options of mapping words into titles of \mathfrak{W} page. We remove (or not) from the set of possible page those of them that do not match in an exact way. The option “exact matching” worked slightly better (although not significantly), so we present it. Then we present individual classifiers followed by the ensembles of classifiers. Subsequent results are for heterogeneous committee.

5.5 Classification for data with the semantic gap

The second experiment focuses on the problem of semantic gap which is observed in classification of data from different domains. For such data often two documents express the same concepts, but as they use different wording (because of existing of synonyms, hypernyms, hyponyms), the conventional classification / clustering algorithms, based on standard bag-of-words approach, do not work well. Such classifiers often do not recognize different linguistic representations for test and training set. Some works relating to the problem were presented in Section 2. Our approach, thoroughly presented above, is different from them.

There are other linguistic phenomena such as ellipsis, paraphrase and other. We focus on synonyms, hypernyms, hyponyms because of Wikipedia structure on which our algorithm is based. We deal with hyper-/hyponyms relation because of \mathfrak{W} category graph structure we operate on. This graph is built on these kinds of relations.

With synonym relation we cope during the phase of mapping words/phrases from the text into \mathfrak{W} pages. The string set attached to a single \mathfrak{W} page contains the page title and all its synonyms. They are extracted from all names of disambiguation pages that point to this particular page.

For experimental design (see Table 3) we used standard classification methods in different settings. As an input for them we used: 1. *terms* – terms from the document; 2. *categories* – categories for a given document produced by SemCat; 3. *concepts* – set of disambiguated concepts (\mathfrak{W} page id) produced during SemCat algorithm.

In Table 4 we present SemCla, ensembles and the heterogeneous committee with semantic classifier.

6 Results

6.1 Classical task

As can be seen in Tables 1 the best method among the considered SemCat algorithms is the one where upon mapping of terms/phrases to \mathfrak{W} pages the ranking of pages corresponding to a term is computed and all of them are taken into account using appropriate weights. The version using only unambiguous terms and phrases has the poorest performance. Modifications of the base method (variants of fitting, shifting the stage of category projection) do not lead to significant changes in performance.

Though SemCla outperforms individual non-semantic classifiers, one can see that a classical classifier ensemble is able to outperform SemCla.

Therefore we turned to considering the impact of inclusion of SemCat into an ensemble of classical classifiers.

The size of the ensemble (25x Balanced Winnow + 25x Bayes) guarantees the stability of the results under various selections of the random training samples.

Experimental settings included: various levels of \mathfrak{W} category graph used to create training samples [Level=1, 2, ∞] as well as various sample sizes per category [$S = 50, 100, 200$]. Optimal results (presented in Tables 1, 2) were achieved for Level=2 and $S = 200$. In particular, Level= ∞ led to noticeably worse performance, since \mathfrak{W} documents selected in the random sample were vaguely related to the desired topic (category).

On the other hand, in every investigated case, results for the Level=1 were worse than for Level=2, since the randomization of the sample for each instance of the classifier was too low (the number of the \mathfrak{W} documents on level 1 was not sufficient to make a sample).

Ensemble of classical classifiers was extended with SemCat (Table 2) using various weights for 1st, 2nd & 3rd category in the SemCat ranking. This setting required further investigation, but usually weights 14/10/6 led to the best classification results. Higher weights caused worse results. Extended ensemble 25x Balanced Winnow + 25x Bayes + SemCat with Level=2, $S = 200$ and weights 14/10/6 usually was the optimal setting (with an exception for shortest documents).

Further extension of the ensemble with LLDA classifier did not improve the results, both in the case of base ensemble (25x Balanced Winnow + 25x Bayes) and the semantic ensemble that included SemCat algorithm.

Presented experiments lead to the following conclusions: the best results were achieved for ensembles that beside standard classification methods (25xBayes + 25xBalanced Winnow) a semantic method was included (either SemCat or SemCla algorithm). Surprisingly, adding more varied set of standard classification methods (Naive Bayes, Winnow and LLDA) did not improve quality of the ensemble.

Ensemble of 25xSemCla classifiers in most cases does not perform significantly better than a single SemCla. It is mainly due to low variance of the individual voting methods within the ensemble.

Table 1. Average values of various precision measures for *DMOZ small* dataset. Parameter L stands for a level of \mathfrak{W} documents used for training sample, S is a sample size per each group of documents. $25x(B,W)$ stands for an ensemble of 25 Bayes an 25 Balanced Winnow classifiers. Vector of numbers that follows the SemCat represents weights attached to top-3 categories inserted into committee.

Method	Description	Lin precision			Precision		
		short	medium	long	short	medium	long
SemCat method	SemCat algorithm with disambiguation algorithm	0.413	0.468	0.553	0.390	0.442	0.531
SemCat method	SemCat : no disambig. concept (pages) weighted using their rank $1/2^i$	0.417	0.463	0.538	0.393	0.436	0.516
SemCat method	SemCat : no disambig. concept (pages) weighted using their rank $1/i$	0.413	0.464	0.553	0.390	0.437	0.531
SemCat method	SemCat : no disambig. concept (pages) weighted uniformly	0.409	0.442	0.513	0.386	0.416	0.492
Classifier	Balanced Winnow (avg of 25) $L=2$ $S=200$	0.547	0.665	0.712	0.488	0.616	0.669
Classifier	Bayes (avg of 25) $L=2$ $S=200$	0.381	0.473	0.572	0.282	0.394	0.503
Classifier	LLDA (avg of 25) $L=2$ $S=200$	0.437	0.553	0.694	0.385	0.505	0.652
Classifier	SemCla (avg of 25)	0.558	0.638	0.698	0.508	0.589	0.654
Ensemble	$25x(B,W)$ $L=1$ $S=50$	0.558	0.652	0.682	0.434	0.558	0.602
Ensemble	$25x(B,W)$ $L=1$ $S=100$	0.540	0.621	0.667	0.415	0.516	0.578
Ensemble	$25x(B,W)$ $L=1$ $S=200$	0.503	0.619	0.672	0.379	0.519	0.578
Ensemble	$25x(B,W)$ $L=2$ $S=50$	0.577	0.684	0.722	0.515	0.65	0.68
Ensemble	$25x(B,W)$ $L=2$ $S=100$	0.578	0.698	0.731	0.507	0.655	0.68
Ensemble	$25x(B,W)$ $L=2$ $S=200$	0.598	0.699	0.753	0.518	0.656	0.711
Ensemble	$25x$ SemCla $L=2$ $S=200$	0.556	0.637	0.694	0.511	0.59	0.648
Ensemble	$25x(B, W, \text{SemCla})$ $L=2$ $S=200$	0.595	0.718	0.787	0.544	0.685	0.758
Ensemble	$25x(B,W)$ $L=2$ $S=200$ + LLDA: 5.0	0.572	0.689	0.748	0.500	0.646	0.711
Ensemble	$25x(B,W)$ $L=2$ $S=200$ + LLDA: 10.0	0.566	0.693	0.748	0.496	0.653	0.711
Ensemble	$25x(B,W)$ $L=2$ $S=200$ + LLDA: 15.0	0.565	0.685	0.744	0.500	0.646	0.703
Ensemble	$25x(B,W)$ $L=2$ $S=200$ + LLDA: 20.0	0.545	0.666	0.740	0.485	0.624	0.695

6.2 Semantic gap problem

As visible in Tables 3, 4 in case of the semantic gap problem, semantic methods and committees lead to much better results than traditional classifiers, even if the latter are operating on the modified representation (bag of categories instead of bag of words).

It can be seen that the usage of terms alone gives poor results when semantic gap occurs. Classical methods are most helped if categories are provided for the training purposes, but the usage of concepts is only half the way as good. This means actually that our SemCla algorithm uses a much deeper insight into the document content than just a category label assignment.

It is also worth to stress the fact that however SemCla (contrary to SemCat) is supervised, it can also be used in unsupervised version. For such a setting, instead of using unobservable document labels as training classes (cf. Figure 2), one can use document clusters, where clustering is also based on the semantic categorization (SemCat algo-

Table 2. Average values of various precision measures for *DMOZ small* dataset. Parameter L stands for a level of \mathfrak{W} documents used for training sample, S is a sample size per each group of documents. $25x(B,W)$ stands for an ensemble of 25 Bayes and 25 Balanced Winnow classifiers. Vector of numbers that follows the SemCat represents weights attached to top-3 categories inserted into committee.

Method	Description	Lin precision			Precision		
		short	medium	long	short	medium	long
Heterogen.	$25x(B,W)$ $L=2$ $S=50$ +SemCat:(7,5,3)	0.626	0.715	0.74	0.577	0.715	0.734
Heterogen.	$25x(B,W)$ $L=2$ $S=50$ +SemCat:(10.5,7.5,4.5)	0.638	0.732	0.759	0.544	0.700	0.719
Heterogen.	$25x(B,W)$ $L=2$ $S=50$ +SemCat:(14,10.6)	0.622	0.742	0.771	0.537	0.652	0.711
Heterogen.	$25x(B,W)$ $L=2$ $S=50$ +SemCat:(17.5,12.5,7.5)	0.610	0.732	0.770	0.522	0.620	0.672
Heterogen.	$25x(B,W)$ $L=2$ $S=100$ +SemCat:(7,5,3)	0.621	0.735	0.784	0.577	0.718	0.773
Heterogen.	$25x(B,W)$ $L=2$ $S=100$ +SemCat:(10.5,7.5,4.5)	0.645	0.746	0.791	0.588	0.695	0.766
Heterogen.	$25x(B,W)$ $L=2$ $S=100$ +SemCat:(14,10.6)	0.634	0.752	0.805	0.559	0.671	0.727
Heterogen.	$25x(B,W)$ $L=2$ $S=100$ +SemCat:(17.5,12.5,7.5)	0.645	0.744	0.809	0.522	0.629	0.667
Heterogen.	$25x(B,W)$ $L=2$ $S=200$ +SemCat:(7,5,3)	0.635	0.731	0.777	0.577	0.722	0.742
Heterogen.	$25x(B,W)$ $L=2$ $S=200$ +SemCat:(10.5,7.5,4.5)	0.645	0.754	0.777	0.581	0.725	0.758
Heterogen.	$25x(B,W)$ $L=2$ $S=200$ +SemCat:(14,10.6)	0.644	0.757	0.780	0.562	0.688	0.765
Heterogen.	$25x(B,W)$ $L=2$ $S=200$ +SemCat:(17.5,12.5,7.5)	0.488	0.570	0.619	0.426	0.508	0.586
Heterogen.	$25x(B,W)$ $L=2$ $S=200$ + SemCat:(10.5,7.5,4.5) + LLDA: 10.0	0.623	0.718	0.776	0.581	0.712	0.750
Heterogen.	$25x(B,W)$ $L=2$ $S=200$ + SemCat:(10.5,7.5,4.5) + LLDA: 15.0	0.591	0.703	0.767	0.555	0.689	0.758
Heterogen.	$25x(B,W)$ $L=2$ $S=200$ + SemCat:(10.5,7.5,4.5) + LLDA: 20.0	0.57	0.689	0.751	0.537	0.679	0.742

rithm) and applies semantic similarity measures defined in Section 3.2. We are going to investigate this direction more deeply in the future, since it has a big advantage in cases where document labels are unavailable and training set cannot be created (e.g. collections of web pages).

7 Conclusions

In this paper we demonstrated the value of semantic approach to the task of document classification. In particular we show here that an unsupervised approach to the classification is possible when using semantic approach. This may be considered as an interesting result by itself. Acknowledgedly, the semantic classifier we introduce does not perform as well as ensembles of traditional classifiers but apparently an inclusion of a semantic categorizer into such an ensemble is capable of significant improvement of its performance in classic classification tasks.

Intuitively, one would imagine that a classifier incorporating semantic information should be superior to traditional classifiers that do not use such information. As we see from our experiments it is not that obvious. Though semantic classifier proved to be a competitor for individual classic classifiers, ensembles of classic classifiers can beat it. Therefore, exploitation of advantages of semantic information requires some level of sophistication and cannot be considered as obvious.

Table 3. Average values of precision measure for classical methods: Bayes (B), Balanced Winnow (W).

		Classification		
		terms	categories	concepts
Bankier: Business Biznes	Bayes	0.397	0.634	0.376
	Winnnow	0.367	0.546	0.323
Forsal: Currencies	Bayes	0.602	0.910	0.620
	Winnnow	0.720	0.870	0.498
Forsal: Finances	Bayes	0.847	0.952	0.814
	Winnnow	0.832	0.874	0.695
Gynecology	Bayes	0.404	0.505	0.233
	Winnnow	0.074	0.205	0.219
Cardiology	Bayes	0.782	0.746	0.502
	Winnnow	0.350	0.438	0.427
Oncology	Bayes	0.758	0.824	0.526
	Winnnow	0.227	0.627	0.390

Table 4. Average values of precision measure for: “semantic classification” (SemCla), ensemble of SemCla, ensemble of Bayes (B), Balanced Winnow (W) and for the heterogeneous committee.

	SemCla	25xSemCla	25x(B,W)	25x(B,W,SemCla)
Bankier (Business Biznes):	0.752	0.830	0.789	0.855
Forsal (Currencies):	0.972	0.983	0.995	0.999
Forsal (Finances):	0.979	0.986	0.965	0.986
Gynecology	0.842	0.844	0.732	0.833
Cardiology	0.900	0.891	0.895	0.916
Oncology	0.868	0.879	0.856	0.904

What is still more important, the semantic classifier turns out to be superior to classical approaches to classification in case of semantic gap between the training data and the data for which the classifier is to be applied. This fact opens up really new horizons for application of machine learning methods in classification of documents in cases e.g. of mergers between various corporations where the local culture leads usually to development of specific languages different between the firms.

This research opens up a number of further interesting areas of research. Semantic approach (in its base, unsupervised setting) could be tested also for clustering tasks under semantic gap scenario as well as to mixtures of classification and clustering.

References

1. Ciesielski, K., Borkowski, P., Kłopotek, M.A., Trojanowski, K., Wysocki, K.: Wikipedia-based document categorization. In: Security and Intelligent Information Systems, SIIS 2011, Warsaw, Poland, June 13-14, 2011. (2011) 265–278
2. Sebastiani, F.: Machine learning in automated text categorization. *ACM Comput. Surv.* **34**(1) (March 2002) 1–47
3. Sebastiani, F.: Text categorization. In: Text Mining and its Applications to Intelligence, CRM and Knowledge Management, WIT Press (2005) 109–129

4. Seppänen, J.K., Bingham, E., Mannila, H.: A simple algorithm for topic identification in 0-1 data. In Lavrac, N., Gamberger, D., Blockeel, H., Todorovski, L., eds.: PKDD. Volume 2838 of Lecture Notes in Computer Science., Springer (2003) 423–434
5. Medelyan, O., Witten, I.H., Milne, D.: Topic indexing with wikipedia. In: Proceedings of the first AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI'08). (2008)
6. Milne, D., Witten, I.H.: An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In: Proceedings of the first AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI'08). (2008)
7. Milne, D.N., Witten, I.H.: Learning to link with wikipedia. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, Napa Valley, CA, USA, October 26-30, 2008, ACM (2008) 509–518
8. Mihalcea, R., Csomai, A.: Wikify!: linking documents to encyclopedic knowledge. In: Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, CIKM 2007, Lisbon, Portugal, November 6-10, 2007, ACM (2007) 233–242
9. Wroblewska, A., Sydow, M.: Debora: Dependency-based method for extracting entity-relationship triples from open-domain texts in polish. In Chen, L., Felfernig, A., Liu, J., Ras, Z., eds.: Foundations of Intelligent Systems. Volume 7661 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2012) 155–161
10. Ramakrishna Murty, M., Murthy, J., Prasad Reddy, P., Satapathy, S.: A survey of cross-domain text categorization techniques. In: Recent Advances in Information Technology (RAIT), 2012 1st International Conference on, IEEE (2012) 499–504
11. Wang, P., Domeniconi, C., Hu, J.: Using wikipedia for co-clustering based cross-domain text classification. In: Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on, IEEE (2008) 1085–1090
12. Nguyen, C.T.: Bridging semantic gaps in information retrieval: Context-based approaches. ACM VLDB **10** (2010)
13. Rafi, M., Hassan, S., Shaikh, M.S.: Content-based text categorization using wikilogy. CoRR **abs/1208.3623** (2012)
14. Pirrò, G., Seco, N.: Design, implementation and evaluation of a new semantic similarity metric combining features and intrinsic information content. In: On the Move to Meaningful Internet Systems. Volume 5332 of LNCS., Springer (2008) 1271–1288
15. Aas, K., Eikvil, L.: Text categorisation: A survey. Report No. 941 (June 1999) ISBN 82-539-0425-8.
16. Grove, A.J., Littlestone, N., Schuurmans, D.: General convergence results for linear discriminant updates. Mach. Learn. **43**(3) (June 2001) 173–210
17. Littlestone, N.: Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. Machine Learning **2** (1988) 285–318
18. Rosenblatt, F.: The perceptron: A perceiving and recognizing automaton. Technical Report 85-460-1, Ithaca, New York (January 1957)
19. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3** (March 2003) 993–1022
20. Steyvers, M., Griffiths, T.: Probabilistic topic models. In Landauer, T., Mcnamara, D., Dennis, S., Kintsch, W., eds.: Latent Semantic Analysis: A Road to Meaning. Laurence Erlbaum (2006)
21. Ramage, D., Hall, D., Nallapati, R., Manning, C.D.: Labeled lda: a supervised topic model for credit attribution in multi-labeled corpora. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1. EMNLP '09, Stroudsburg, PA, USA, Association for Computational Linguistics (2009) 248–256
22. Blei, D.M., McAuliffe, J.D.: Supervised topic models. In: NIPS. (2007)

23. Ramage, D., Manning, C.D., Dumais, S.: Partially labeled topic models for interpretable text mining. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. KDD '11, New York, NY, USA, ACM (2011) 457–465
24. Hastie, T., Tibshirani, R., Friedman, J.H.: The Elements of Statistical Learning. Springer (2008)